

EECE.2160: ECE Application Programming

Spring 2016

Lecture 28: Key Questions

April 11, 2016

For today's exercise, you will complete the following functions that work with the structures `Name` and `StudentInfo`. The structure definitions are listed below:

```
typedef struct {
    char first[50];
    char middle;
    char last[50];
} Name;
```

```
typedef struct {
    Name sname;
    unsigned int ID;
    double GPA;
} StudentInfo;
```

The function descriptions are as follows:

For the `Name` structure:

- **void printName(Name *n)**: Print the name pointed to by `n`, using format `<first> <middle>. <last>` *(completed 4/8)*
- **void readName(Name *n)**: Prompt for and read a first, middle, and last name, and store them in the structure pointed to by `n` *(completed 4/8)*

For the `StudentInfo` structure:

- **void printStudent(StudentInfo *s)**: Print information about the student pointed to by `s` *(completed 4/8)*
- **void readStudent(StudentInfo *s)**: Prompt for and read information into the student pointed to by `s` *(completed 4/8)*
- **void printList(StudentInfo list[], int n)**: Print the contents of an array `list` that contains `n` `StudentInfo` structures
- **int findByLName(StudentInfo list[], int n, char lname[])**: Search for the student with last name `lname` in the array `list`. Return the index of the structure containing that last name, or -1 if not found
- **int findByID(StudentInfo list[], int n, unsigned int sID)**: Search for the student with ID # `sID` in the array `list`. Return the index of the structure containing that last name, or -1 if not found

From StudentInfo.c:

```
// Print list of students
void printList(StudentInfo list[], int n) {

}

// Find student in list, based on last name
// Returns index if student found, -1 otherwise
int findByLName(StudentInfo list[], int n, char lname[]) {

}

}
```


3. Explain the `calloc()` function.

4. Explain the `realloc()` function.